

Aprendizaje Deductivo, Inductivo, y Abductivo con Algoritmos Genéticos

José Alejandro Brito Boadas
jabbmv@cantv.net

Postgrado en Computación, Universidad de Los Andes, Venezuela

Resumen Este artículo discurre sobre la implementación de tres mecanismos de aprendizaje con algoritmos genéticos, en concreto, aprendizajes deductivo, inductivo y abductivo. En primer lugar, se presenta una revisión del estado del arte en este tópico. Luego, el énfasis se traslada al planteamiento de un modelo teórico de aprendizaje basado en la representación de la realidad de los agentes mediante cláusulas de la lógica de primer orden. Por consiguiente, el trabajo también constituye una muestra de la conjunción de algoritmos genéticos y la lógica de predicados. Al final, se discuten someramente algunos resultados prácticos obtenidos, que verifican positivamente la factibilidad de la propuesta.

1. Introducción

Los Algoritmos Genéticos (en lo sucesivo, AGs) representan una de las técnicas inteligentes más utilizadas en la actualidad, principalmente por su versatilidad y notables resultados prácticos. Uno de los temas que más atención ha recibido por parte de la comunidad científica que experimenta con tales técnicas es el Aprendizaje Artificial, disciplina que estudia el proceso de aprendizaje en general, con énfasis en la traslación a las máquinas de lo hasta ahora conocido sobre dicho proceso en los seres vivos. El problema fundamental es que los agentes artificiales con frecuencia deben operar en ambientes no deterministas, dinámicos, y en general, bastante complejos, por lo que no resulta factible incorporar a priori todo el conocimiento requerido para un rendimiento aceptable; en ocasiones, tal conocimiento ni siquiera se encuentra disponible con antelación. Por ende, resulta imperativo que el agente *aprenda* sobre el ambiente, utilizando las nuevas percepciones y el conocimiento que posee. Dependiendo de los mecanismos de inferencia a los que recurra el agente, el proceso puede catalogarse como deductivo, inductivo, o abductivo. En este contexto, el AG aporta su gran potencial en el manejo de poblaciones de deducciones, observaciones e hipótesis, con las ventajas del paralelismo implícito, y del refinamiento y descubrimiento de información.

2. Marco Teórico

Los AGs constituyen un conjunto de modelos matemáticos inspirados en la teoría Darwinista de la evolución. En concreto:

Un AG es un método de búsqueda (ciega) de soluciones, cuyo funcionamiento consiste en *evolucionar* una población de potenciales soluciones (individuos) de un problema, con el objetivo de que las soluciones mejor adaptadas prevalezcan y transmitan sus características a soluciones hijas. La población se transforma mediante la aplicación de operadores genéticos. Así, luego de una sucesión suficiente de generaciones, debería disponerse de un grupo de soluciones bastante adaptadas al problema particular.

En este documento, los AGs se abordan en cuanto a su relación con el aprendizaje artificial. Según Mitchell [27]:

Un agente artificial **aprende** de la experiencia E con respecto a alguna clase de tareas T y medida de rendimiento P, si su desempeño en las tareas de T, según la medida P, mejora con la experiencia E.

La conjunción de ambas nociones suele denominarse Aprendizaje Genético [20]. Aquí conviene aclarar que los individuos codifican conocimiento sobre el ambiente, en la forma de hipótesis o de hechos ciertos. La noción de certidumbre siempre se encuentra en relación con la base de conocimientos (en lo sucesivo, BC) y el procedimiento de demostración del agente. Esta relación determina tres niveles epistemológicos que, de acuerdo con los trabajos de Pierce [43], permiten identificar al proceso de aprendizaje como:

- *Deductivo*: Esta clase de inferencia en realidad no aporta información semántica a la BC, por cuanto conduce de una generalización a una particularización. Por tal razón, con frecuencia se encuentra asociada a otros paradigmas de aprendizaje. Operadores como el de clonación constituyen una reexpresión de la sentencia deductiva $A \supset A$, una forma simple de probar algo, pues permite preservar el conocimiento de una generación a otra. Y, si sólo una parte de la información es transmitida al hijo, se estaría siguiendo un enfoque descendente en el que de la generalización del padre se arriba a información específica (parcial) en el hijo. Por otra parte, algunas investigaciones recientes recurren a las facultades deductivas de los AGs con fines de modelado de la deducción humana [8,21]. En resumen, la deducción se expresa [30]:

Si se aceptan $p(1)$ y $\forall X(p(X) \supset q(X))$, entonces, por *deducción*, concluir $q(1)$.

En los AGs, la transmisión de información a los hijos por parte de los progenitores constituye un proceder deductivo, por cuanto el conocimiento codificado en los progenitores es recibido por el hijo íntegro o parcialmente [14].

- *Inductivo*: Si el AG se utiliza para aprender teorías a partir de hechos observados, entonces se trata de aprendizaje genético inductivo. En

este caso, se busca obtener individuos que, dependiendo de la información que codifiquen, resulten más generales que sus predecesores, donde la relación de generalidad dependerá del problema. Formalmente [30],

Si se observan los hechos $p(1)$, $p(2)$, $p(3)$, y los únicos objetos(constantes) del mundo son 1, 2 y 3, entonces, por *inducción*, concluir $\forall X p(X)$.

- *Abductivo*: En este caso, la capacidad de exploración del AG se conjuga con la explotación de nuevas hipótesis explicativas de las observaciones. Por lo general, esto implica que el sistema de aprendizaje genético construye hipótesis cuya validez comprueba un evaluador. Es el razonamiento de los diagnósticos, que tratan de proporcionar alguna explicación a los síntomas o acontecimientos del ambiente. A diferencia de la deducción, aquí se infiere la causa a partir del efecto:

Si se sabe que $\forall X (p(X) \supset q(X))$, al observar $q(1)$, puede proponerse la hipótesis $p(1)$ para explicar la ocurrencia de $q(1)$.

A continuación, se expone una selección de trabajos representativos y recientes en el área de Aprendizaje Genético, bajo las tres ópticas descritas.

3. Revisión Bibliográfica

3.1. Aprendizaje Genético Deductivo

En [18] se utiliza el AG para la generación evolutiva de rostros, con aplicaciones inmediatas en la Criminalística, específicamente, en las reconstrucciones faciales. La población inicial es un conjunto de imágenes faciales con los rasgos deseados, y a través de la evolución, se crean nuevos rostros que combinan los rasgos transmitidos por los progenitores. En este mismo contexto, resulta posible agilizar la búsqueda si se dispone de información que pueda indicar al AG cuáles imágenes deberían combinarse con preferencia. El proceder es deductivo por cuanto se trata de obtener nuevas imágenes a partir de la combinación de la información más general representada por las imágenes de la población inicial. Esto es precisamente lo que se hace en [3], donde se incorporan restricciones geológicas y geométricas a los operadores genéticos, para la interpretación de datos sísmicos. En ambos casos, las imágenes construidas deductivamente han resultado muy útiles en la práctica.

Por otra parte, en el ámbito de la Lingüística Computacional, Bianchi y Delmonte [4] emplean el aprendizaje deductivo para resolver la cuestión de la ambigüedad de palabras en textos en lenguaje italiano, y para la conversión de un texto en una secuencia de categorías gramaticales. En este caso, las poblaciones representan árboles de categorías gramaticales, y a través de modificaciones a los mismos, también se trata de inducir nuevas reglas de clasificación gramatical. Precisamente, esta combinación de deducción e inducción representa uno de los aspectos más interesantes de este estudio.

Una muy reciente aplicación de los AGs ha sido hacia el aprendizaje deductivo y abductivo de las relaciones en las organizaciones humanas, mucho más complejas que las artificiales. Dicha propuesta recibe el nombre de *Human-Based Genetic Algorithm* (HBGA) [21,22]. En síntesis, se trata de un enfoque evolutivo del modelado organizacional. A diferencia del enfoque netamente abductivo, el HBGA confiere al humano los roles tanto de crítico como de creador, y todos los operadores genéticos se desarrollan con la intervención de las personas. Se parte con un modelo, y a través de la evolución, el propio AG y los miembros de la organización aprenden sobre la misma, considerando las definiciones a priori del modelo (deducción), y las nuevas hipótesis de funcionamiento organizacional (abducción).

Una perspectiva distinta de modelado, basada en el aprendizaje netamente deductivo, es desarrollada por Chakraborti [8]. En primer lugar, la investigación reconoce que la noción de *Validez* constituye un nodo central de todas las ciencias. Sin embargo, aún no existe una descripción precisa de la manera en que se desarrolla la deducción humana. Por ejemplo, ¿cómo el ser humano *computa* realmente una consecuencia a partir de un conjunto de premisas dadas?. En este sentido, se plantea que el razonamiento deductivo puede concebirse como la búsqueda evolutiva de la conclusión a partir de las premisas. El resultado principal del estudio es que los AGs representan una herramienta excelente para la deducción.

3.2. Aprendizaje Genético Inductivo

Enmarcado en la disciplina de Lingüística Computacional, Echizenya et al [13] presentan un estudio para la traducción al inglés de textos en japonés. Lo relevante es que el AG crea las instancias para la inducción de las reglas de traducción (a diferencia de los enfoques tradicionales, en los que las instancias de aprendizaje se definen previamente).

Los AGs también han sido combinados profusamente con otras técnicas de Computación Inteligente con el propósito de extender el alcance del proceso de aprendizaje. Una de tales hibridizaciones recibe el nombre de Aprendizaje Genético Difuso [9,15], y conjunta reglas difusas con la inferencia evolutiva. Así, en [9] se emplean para extender un sistema clasificador (LCS) con reglas difusas, y múltiples funciones objetivo, en el contexto de sistemas de bases de datos. No obstante, el aprendizaje inductivo de este tipo de sistemas se dificulta por la aparición de un elevado número de características que incrementan la dimensión del problema. La dificultad surge del crecimiento exponencial del espacio de búsqueda de reglas difusas. Para abordar esta cuestión, en [7] se presenta un proceso de selección genética de características que puede integrarse en un sistema genético de aprendizaje multietapas. El estudio destaca por el uso de las características, que permiten reducir el espacio de búsqueda, y por los resultados en cuanto a la descripción de sistemas con reglas difusas. Otra alternativa se propone en [15], donde el espacio de búsqueda se incrementa gradualmente. Por último, con una orientación más práctica, se han creado también LCS Genético Difusos para la detección de intrusos en una red, con altas tasas de reconocimiento y autonomía [10,16].

Por su parte, la combinación de Redes Neuronales Artificiales y AGs recibe el nombre de Neuroevolución, cuyo problema típico es la determinación del número de capas y nodos, tarea en la que el AG puede desenvolverse como una buena heurística [37]. Las investigaciones son muy diversas. Por ejemplo, en [36] se utilizan exitosamente para el reconocimiento de imágenes satelitales. Por otro lado, Cantú-Paz y Chandrika crearon un sistema clasificador de galaxias con neuroevolución, a partir de observaciones astronómicas. Uno de los logros del estudio es la demostración de que el AG puede obtener la mejor topología para la red y entrenarla [6]. En [5] el modelo se emplea para el refinamiento de RNAs expertas, que son redes que no sólo computan la respuesta, sino también un estimado de lo correcta que es dicha respuesta. En [45] se usa un enfoque evolutivo con las RNA para ayudar a mejorar la tolerancia a fallos de las redes en cascada. Los resultados también muestran un incremento en la capacidad de generalización de la red. Por último, McQuesten[26] propone una extensión a la Neuroevolución basada en la *cultura*. El proceso de cultura es la transmisión de comportamiento de una población a otra, a través de algún mecanismo no genético. La cultura proporciona enormes ventajas para la supervivencia en el mundo biológico, y al trasladar el concepto a la neuroevolución, se demuestra que es posible obtener mejoras en la velocidad de aprendizaje y en la calidad de la solución, con el agregado adicional de una operación sin cambios en la función de adaptación.

Con una perspectiva más teórica, otros estudios analizan las cualidades de la neuroevolución con redes de topología variable [37,38,40]. La mayoría de los métodos de neuroevolución emplea redes de topología fija. En las investigaciones referidas, se estudia la evolución sin esta restricción, con la finalidad de corroborar que la topología de la red puede emplearse para incrementar la eficiencia del AG si minimiza la dimensión del espacio de pesos. Los resultados han sido excelentes, y representan una contribución de la neuroevolución a los AG, al confirmar que la mejor evolución transcurre no sólo a través del relevo generacional, sino considerando también la *complejificación*¹ de las soluciones, permitiendo obtener soluciones más complejas, e incrementando la analogía con el proceso biológico. Estudios sucesivos sobre esta complejificación con AGs se presentan en [39,41,42].

En el área de Robótica, Augustsson et al [2] presentan una aplicación novedosa de las técnicas evolutivas, al incorporar un sistema de aprendizaje genético a un robot volador. Con este tipo de máquinas, uno de los principales requisitos es un aprendizaje en línea, pues se desean reacciones lo más inmediatas posibles. Otra idea que sustenta el proyecto es que pequeños organismos voladores, como el mosquito, no poseen una capacidad inteligente amplia y sin embargo, son capaces de resolver problemas complejos. Lo que la investigación demuestra es la posibilidad de incorporar a un pequeño robot volador un sistema evolutivo lineal capaz de aprender a desempeñarse correctamente a partir de percepciones aerodinámicas. También resalta el trabajo de DiPietro et al [12] sobre la resolución del problema *Keepaway*, referido a la inducción de la mejor

¹ Complexification

respuesta de atacantes ante defensores en el soccer robótico. Esta investigación comprueba las cualidades de los AGs en labores de planificación. Finalmente, se citan dos investigaciones que combinan deducción e inducción para el diseño de circuitos electrónicos. Al comienzo, los sistemas conocen los requisitos funcionales de los circuitos, y tal información se transmite deductivamente a los relevos generacionales. Así, en [19] se recurre al AG para el diseño de hardware evolutivo (reajustarse cuando cambien las especificaciones funcionales), mientras en [44] lo utilizan para aprender inductivamente a simplificar un diseño de circuito cuántico.

3.3. Aprendizaje Genético Abductivo

Inicialmente, se revisa el aprendizaje abductivo en el ámbito musical, en el que las composiciones se abordan bajo la figura de hipótesis (de preferencias musicales). Por ejemplo, en [25] se presenta un ambiente virtual complejo, habitado por *agentes sónicos*, con capacidad para oír y generar sonidos. El sistema mantiene retroalimentación con una audiencia humana, y los agentes evolucionan para conservar lo más que puedan el interés de dicha audiencia. El resultado inmediato de la evolución es la hipótesis que cada agente ofrece sobre las preferencias musicales de la audiencia, codificadas bajo la forma de reglas muy complejas. Además, la investigación cambia el proceder Darwiniano por el Lamarckismo, para acelerar la obtención de hipótesis musicales satisfactorias. Por otra parte, en esta área también se han planteado variantes al AG, como el IGA², en el proceso de composición musical sobre la plataforma Vox Populi [24], y el CBGA³ [33], donde también se combina la deducción para crear nuevas piezas musicales (construcciones) a partir de las existentes. Una investigación similar, con preponderancia de la interacción humana para la revisión de las hipótesis musicales es [28].

Por otra parte, Hamel combina Programación Lógica Inductiva [30], Aprendizaje de Conceptos, Programación Genética y Lógica de Ecuaciones, para definir una nueva área, la Programación Lógica Ecuacional Inductiva [17]. Estos sistemas proceden a partir de una población de teorías candidatas, cuya adaptación se encuentra supeditada a lo bien que expliquen los ejemplos. Es decir, con el enfoque evolutivo se promueve la subsistencia de las mejores hipótesis.

En [23] presentan un nuevo enfoque para el reconocimiento de regiones promotoras de genes eucarióticos. Se considera un enfoque abductivo por cuanto el AG efectúa una búsqueda de una partición óptima de la región de promotores de los genes de *Drosophila melanogaster*, con el propósito de obtener fragmentos que no se superpongan (hipótesis) para seleccionar las frecuencias significativas de dinucleótidos que aparecen en los mismos. El AG demostró ser muy eficiente comparado con los modelos de cadenas de Markov, RNAs y otras técnicas de análisis discriminante.

² *Interactive Genetic Algorithm*, es una variante del AG empleada para el aprendizaje por refuerzo, donde el ajuste poblacional está supeditado a la apreciación del crítico. Precisamente, el HBGA es, contextualmente, una extensión del IGA.

³ *Construct Based Genetic Algorithm*.

Por último, este año se presentó un sistema de detección de intrusos de nueva generación, basado en aprendizaje abductivo con AG [11]. El sistema incorpora una *matriz causal* que indica, para diversos tipos de ataques, los síntomas que presenta la red. La idea es considerar la red como un ser biológico, que comienza a funcionar mal, y a partir de los síntomas se intenta proponer una explicación de la anomalía. El AG opera sobre poblaciones de hipótesis, con una función de evaluación difusa, que considera como mejores individuos a aquellos que proporcionen una buena explicación sobre el ataque que experimenta la red (de acuerdo con la matriz causal). Los resultados preliminares revelan una detección satisfactoria de subclases de ataques DoS (Denial of Service).

4. Definición del problema

La anterior revisión bibliográfica muestra que el problema del aprendizaje inductivo, abductivo y deductivo no es abordado de forma integral y explícita, y así cada investigación aplica, por separado, alguno de los paradigmas. Adicionalmente, hay que acotar que dichos trabajos tampoco ofrecen unas bases teóricas sólidas, sino que proceden de inmediato a exhibir las bondades del aprendizaje genético en un caso de estudio particular. Con el fin de subsanar estas carencias, la presente investigación desarrolla un modelo teórico para conjuntar los tres enfoques de aprendizaje genético, incorporando también representaciones de la lógica de primer orden y un mecanismo, simple pero suficiente, para el manejo de la incertidumbre. Otras investigaciones también utilizan las representaciones de alto nivel de la lógica de predicados en los AG, como REGAL [31,32] y SIAO1 [1], aunque limitados al aprendizaje inductivo. Nuestro aprendiz artificial es un agente basado en el conocimiento que además de las operaciones típicas DECIR [informa a la BC sobre las nuevas percepciones] y PREGUNTAR [interroga a la BC sobre la acción a ejecutar en un momento dado], añade la operación INFERIR(EVOLUTIVAMENTE) la cual, en cualquier instante, aplica el AG para obtener nuevo conocimiento, y así extender la semántica de la BC. La manera de lograr esto constituye el tema central de la investigación. Por otra parte, la BC empleada debe presentar las siguientes características. Primero, el conocimiento se representa como cláusulas en lógica de primer orden. Es decir, cada elemento de la BC tiene la forma $C_i = \bigvee_{i=1}^n L_i$, donde L_i = literal (positivo o negativo). Se elige esta representación porque, en primer lugar, sus propiedades han sido ampliamente estudiadas, y porque suele ser buena para representar los objetos y relaciones del mundo. Luego, la BC puede considerarse como una conjunción de todas sus cláusulas. El segundo detalle es que esta lógica, sin embargo, muestra dificultades para lidiar con la incertidumbre. En consecuencia, aquí percibiremos a la BC como un conjunto de pares $BC = \{(C_1, \mu_1), (C_2, \mu_2), \dots, (C_n, \mu_n)\}$ donde μ es un grado de certidumbre en $[0, 1]$. μ no es una probabilidad, sino un grado de verdad, similar al grado de membresía en los conjuntos difusos. Las cláusulas con certidumbre 1 o cercanas a 1 son aquellas que representan el conocimiento sobre el mundo con más fidelidad; las cláusulas cuya certidumbre tiende a 0 son inconsistentes o carecen de evidencia.

La asignación de tales certidumbres dependerá de la implementación del agente y de los operadores genéticos, y además, puede cambiar a lo largo de la ejecución del AG.

5. Propuesta de Solución

5.1. Población e Individuos

A nivel de fenotipo, un individuo constituye una BC, es decir, siguiendo la estrategia de Pittsburgh [7], cada individuo se compone de un conjunto de cláusulas. La idea es que la población represente, a su vez, un conjunto de BC's. Inicialmente, se dispone de una sola BC, BC_O , la cual encierra el conocimiento del agente sobre el mundo, en un momento dado. Luego, las BC's de la población inicial se derivan de BC_O aplicando mecanismos de inferencia (deductiva, inductiva y abductiva) sobre las cláusulas de ésta, al azar. El conjunto T_i de cláusulas resultantes en cada aplicación i se une con BC_O para formar los miembros de la población inicial, como se aprecia en la Figura 1.

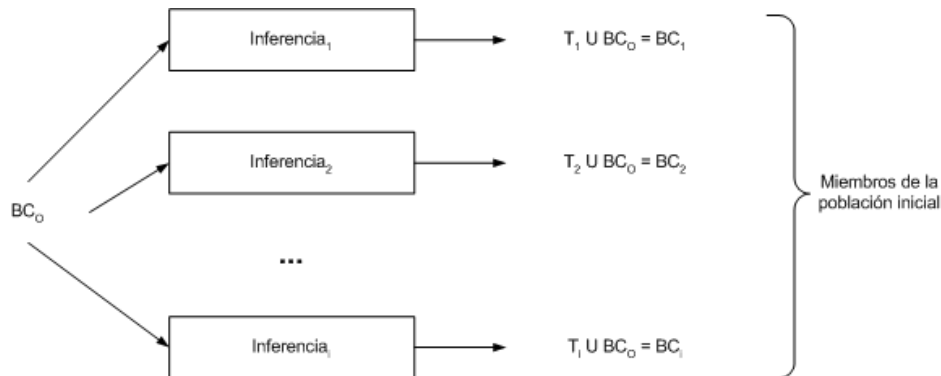


Figura 1. Formación de la población inicial

Posteriormente se describirá la manera de aplicar las inferencias sobre BC_O , apelando a un proceso denominado *emparejamiento*. Definida la población, mediante el relevo generacional se persigue eliminar las redundancias y contradicciones. Empero, un problema que deberán afrontar las implementaciones es que la BC, y por ende la población, poseen un tamaño teóricamente infinito⁴.

Ahora procedamos a discutir la representación a nivel de genotipo. La BC se compone de cláusulas, y éstas, a su vez, de literales. Recordemos que

⁴ Aunque éste no es un problema exclusivo de nuestro modelo. El dilema es, cuánto espacio se necesita para almacenar el conocimiento.

un literal es un predicado (o su negación), definiendo una relación entre términos (variables o funciones). Representar una función puede tornarse excesivamente complejo, por lo que se impondrá una restricción: sólo se admiten funciones de aridad 0, es decir, constantes. Esta restricción acarrea problemas con las listas y la skolemización, pero será suficiente para ilustrar el proceso de aprendizaje genético. No obstante, la ventaja es que el universo de Herbrand U_H de la BC es finito. Por otro lado, las variables tienen ámbito local (de cláusula) por lo que pueden codificarse en cada individuo, a diferencia de U_H y el conjunto de predicados U_P , los cuales se distribuyen entre toda la BC, impidiendo codificaciones locales. Además, teóricamente, individuos, cláusulas y literales pueden tener cualquier cantidad de cláusulas, literales y términos, respectivamente, lo que requerirá un genotipo infinito. En la práctica, habrá que establecer una cota superior a dichas dimensiones, α_C , α_L y α_T . Luego, un individuo es un conjunto de α_C cláusulas; la estructura de la cláusula se muestra en la Figura 2. Allí, **signo** señala si el literal es positivo o negativo, **predicado** es un índice en algún diccionario con U_P , y **término** es, o un índice en algún diccionario con U_H , o una codificación local de las variables. La representación, naturalmente, debe también contener la certidumbre de la cláusula. Por ejemplo, supóngase que $\alpha_L = \alpha_T = 5$, que el 0 (cero) funge como delimitador de literales y términos, y que se desea representar la cláusula $\neg p(X) \vee q(juan, Z) \vee \neg y(X, Z)$, con certidumbre 0,5. La cláusula se aprecia en la Figura 3.

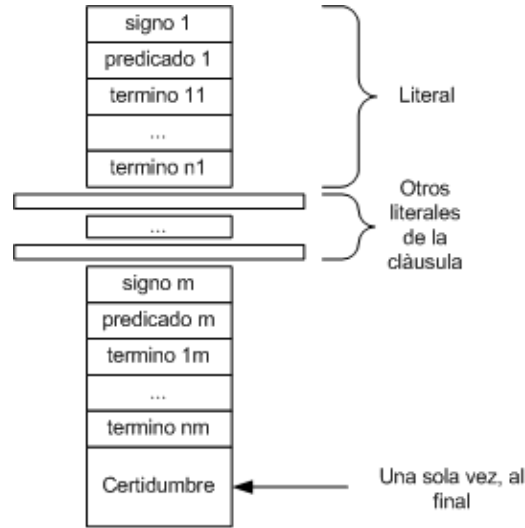


Figura 2. Estructura general de una cláusula, con $\alpha_L = m$ y $\alpha_T = n$

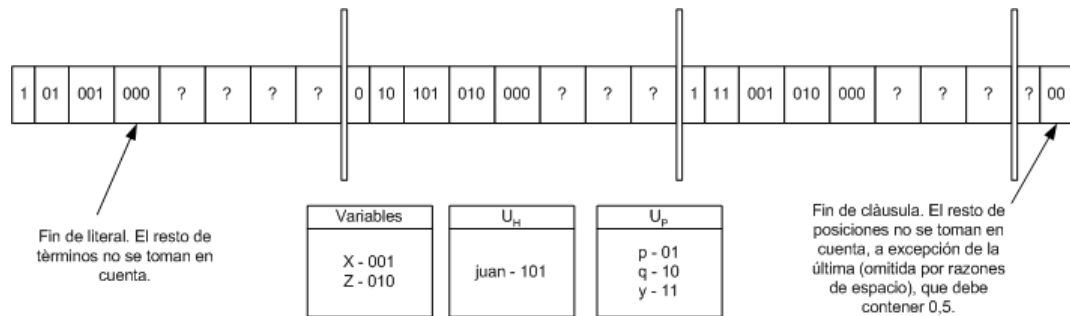


Figura 3. Representación para $\neg p(X) \vee q(juan, Z) \vee \neg y(X, Z)$

5.2. Evaluación y Ajuste

Aplicando deducción, inducción o abducción, puede derivarse conocimiento a partir de BC_0 . En este sentido, la búsqueda de nuevo conocimiento debe estar guiada por un *objetivo*, para así descartar todas las derivaciones irrelevantes en la situación del agente. Por ejemplo, un robot podría plantearse como objetivo el desplazamiento en alguna dirección, y determinar cuál es la más consistente con lo que sabe hasta el momento. Esto, más allá del rendimiento en dicho instante, permitirá ampliar la propia BC_0 para mejorar el desempeño a posteriori. El objetivo debe representarse en forma clausal, en una *cláusula objetivo*. Cada cláusula de una BC se contrasta con el objetivo a fin de determinar si lo contiene. Según esta comparación, la cláusula recibe un valor numérico v_j , como evaluación. En pseudocódigo:

```

real evalua_clausula(CLAUSULA evaluada, CLAUSULA objetivo)
    bono = 0;
    si evaluada contiene objetivo
        dif = total_literales(evaluada) - total_literales(objetivo);

        // para premiar a los individuos que incluyan el objetivo
        si dif == 0
            retorna 1000.0;
        sino
            bono = 2.0;
        fin si
    fin si

    // se premian las clausulas cortas y de buena certidumbre
    si total_literales(evaluada) == 1 y certidumbre(evaluada) == 1.0
        retorna 9.0;
    fin si

    retorna (8.0 / total_literales(evaluada)) + cert + bonus;
fin evalua_clausula

```

Las constantes han sido determinadas empíricamente. Nótese que mientras más similar al objetivo resulte la cláusula evaluada, más alta será su evaluación; si son idénticas se asigna un valor bastante alto (1000.0) para promover su supervivencia durante el proceso de selección. Al mismo tiempo, las cláusulas de menor longitud son más recompensadas, hasta el punto de que si la longitud es mínima (1) y son verdícas, recibirán una evaluación buena. La razón de este proceder es la heurística de *unidad preferente* [34]: cuando se pueda, inferir a partir de cláusulas con un único literal, pues los descendientes tendrán menos literales que la más grande de las cláusulas padres⁵.

Finalmente, la función de evaluación de cada individuo, es decir, de cada BC, viene dada por $f_i = \sum v_j$, para cada cláusula j en la i -ésima BC. Por último, el ajuste es f_i/\bar{f} .

5.3. Selección y Reemplazo

Una implementación puede decantarse en favor de los mecanismos que resulten más convenientes. En la implementación realizada, y que se describirá en una sección venidera, la selección se realiza por sorteo, y el reemplazo es por inserción.

5.4. Operadores

Los operadores son el cruce y la mutación. Se utiliza el cruce clásico, de un solo punto, donde dos progenitores intercambian cláusulas para formar dos descendientes, como se presenta en la Figura 4. Allí se aprecia que el primer descendiente recibe del primer padre una cantidad de cláusulas igual a CP, donde CP es el punto de cruce, seleccionado al azar. Luego, el padre 2 le proporciona la cantidad de cláusulas necesarias para completar una cantidad de cláusulas igual a las del primer padre. Finalmente, las cláusulas restantes de ambos padres conforman al segundo hijo.

La mutación consiste en cambiar, al azar, una variable por algún miembro de U_H , o una constante por una variable, existente o no en la cláusula. Sin embargo, las cláusulas resultantes de la mutación carecen de evidencia que las respalde, por lo que su certidumbre debe corresponder a una penalización de la cláusula original (por ejemplo, $\mu_i = \mu_i/7$). A pesar de todo, la mutación tiene sus ventajas. Por ejemplo, a partir de $p(a) \supset q(a)$ podría obtenerse la cláusula más general $p(X) \supset q(X)$, y viceversa. Así también se incorpora nuevo conocimiento a la BC. Funcionalmente, resulta similar a la mutación en SIAO1, aunque esa propuesta es más amplia al también considerar los predicados dentro de su alcance.

⁵ Aplica sólo a la inferencia deductiva y abductiva. Más adelante se comprobará que la inducción no reduce el tamaño de la descendencia.

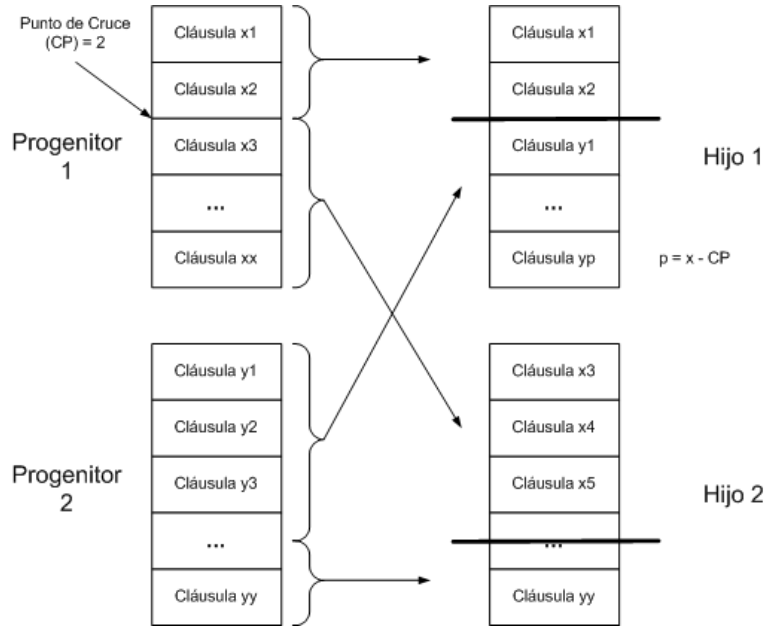


Figura 4. Cruce de dos individuos, con $CP = 2$.

5.5. Formación de la población inicial

Anteriormente se refirió que los individuos de la población inicial incluyen las mismas cláusulas de BC_0 , más nuevas cláusulas resultantes de la inferencia deductiva, inductiva o abductiva. Estas nuevas cláusulas se obtienen a partir de un proceso de *emparejamiento* de las cláusulas en BC_0 . Las cláusulas a emparejar se determinan aleatoriamente en cada caso, para que las BC's derivadas sean diferentes entre sí. Nótese que el emparejamiento actúa a nivel del individuo, y no constituye un operador genético, sino un proceso externo necesario para la inferencia.

Emparejamiento En este modelo teórico del aprendizaje genético, no debería emparejarse cualquier par de individuos. Recordemos que la BC encierra el conocimiento actual sobre el mundo, y por ende, usando los paradigmas de aprendizaje bajo estudio, el emparejamiento de dos cláusulas como $p(a)$ y $q(b)$ difícilmente aporta información adicional, excepto $p(a) \wedge q(b)$, pero esa conjunción ya se encuentra implícita en la BC. Así, pues, hay que tratar de emparejar cláusulas que permitan inferir nuevo conocimiento, pero esta discusión se pospone hasta el final de la sección.

Existen varios tipos de emparejamiento, correspondientes a los paradigmas de aprendizaje considerados. Dadas dos cláusulas padres, C_1 y C_2 , primero debe determinarse el tipo de emparejamiento aplicable, si alguno lo es.

1. **(Emparejamiento deductivo)** El resolvente (descendiente) C_r de C_1 y C_2 es la disyunción de ambos, excepto que si existe un par de literales L_1 y $\neg L_2$, tales que uno de los padres contenga L_1 y el otro L_2 , y si L_1 y L_2 son unificables, entonces L_1 y L_2 se remueven de C_r ⁶. La sustitución θ proveniente de la unificación de L_1 y L_2 se aplica a la disyunción para obtener el C_r definitivo. Esto es lo mismo que hace el proceso de resolución en lógica de primer orden, y constituye un mecanismo deductivo de inferencia.

Ejemplo

C_1 : $\neg p(a) \vee q(b) \vee r(c)$

C_2 : $s(X) \vee p(X)$

C_r : $s(a) \vee q(b) \vee r(c)$

En C_r aparece $s(a)$ y no $s(X)$, porque se aplica la sustitución $\theta = \{X/a\}$. Ahora bien, respecto a la certidumbre del resolvente, μ_r , puede obtenerse a partir de la de los padres (μ_1, μ_2) , usando algún criterio como:

Promedio: Se asume que la certidumbre de C_r se ubica entre la de sus progenitores, $\mu_r = (\mu_1 + \mu_2)/2$.

Mínimo: Este es un enfoque más conservador, que dota a C_r con la mínima certidumbre de los padres, $\mu_r = \min(\mu_1, \mu_2)$.

Nótese que con el emparejamiento deductivo se genera un único descendiente a partir de dos progenitores.

2. **(Emparejamiento abductivo)** Si C_1 es un hecho⁷, y C_2 una regla⁸, y C_1 aparece en el consecuente de C_2 , entonces se generan, como descendientes, los literales en el antecedente de la regla, negados. En la oración anterior, *aparece* significa que C_1 unifica con un literal positivo de C_2 . La sustitución resultante se aplica a cada descendiente. Un ejemplo ayudará a clarificar la explicación.

⁶ L_1 y L_2 son entonces *literales complementarios*.

⁷ Por sencillez, denominaremos *hecho* a una cláusula con un único literal, positivo y sin variables.

⁸ Por sencillez, denominaremos *regla* a cualquier cláusula con al menos un literal positivo y al menos un literal negativo, eg., $p(X) \supset q(X)$, que se reescribe $\neg p(X) \vee q(X)$, es una regla.

Ejemplo

$C_1: r(a)$

$C_2: \neg p(X) \vee \neg q(X) \vee r(X)$

$C_r^1: p(a)$

$C_r^2: q(a)$

$r(a)$ unifica con $r(X)$, dejándonos la sustitución $\{X/a\}$; de allí que como descendientes se tengan $p(a)$ y $q(a)$. El proceder es abductivo porque a partir de la regla se extrae una *explicación* para el hecho. En el ejemplo, la conjunción $p(a) \wedge q(a)$ y la regla (C_2) proporcionan una explicación para $r(a)$. Ahora bien, debe determinarse el grado de certidumbre de los descendientes. Sea $\#L$ la cantidad de literales positivos en C_2 , descontando C_1 . Luego, todos los descendientes tendrán el mismo grado de certidumbre, calculado mediante $(\mu_1 + \mu_2 - \#L)/3$. Se divide entre 3 porque la abducción es más débil que la deducción, y se sustrae $\#L$ del numerador porque mientras más consecuentes haya en la regla, más débil será la hipótesis representada por el antecedente, pues hay más alternativas (ie., literales en el consecuente) para satisfacer la regla.

Nótese que en el emparejamiento abductivo puede generarse más de un descendiente.

3. (**Emparejamiento inductivo**) Se presentan dos formas de este emparejamiento.

Primera Forma

Si C_1 y C_2 son fórmulas básicas (ie., sin variables), y ambas poseen los mismos predicados (en cuanto a signo, nombre, aridad y cantidad), puede intentarse la inducción de una expresión más general, que sería el descendiente C_r , el cual consistiría de los mismos predicados, con los respectivos términos cambiados por un proceso de generalización. Este proceso de generalización consiste en emparejar las constantes de C_1 con sus similares en C_2 , y luego:

```
para cada emparejamiento (term1, term2)
  si term1 = term2,
    pasar term1 al descendiente
  sino
    crear una variable, y pasarla al descendiente. Luego
    reemplazar las otras ocurrencias de (term1, term2),
    en ese orden, por la variable creada.
fin-si
fin-para
```

Nuevamente, el ejemplo ayudará a comprender lo anterior.

Ejemplo

$C_1: \neg p(a, b, c) \vee q(d, c)$

$C_2: q(d, e) \vee \neg p(f, b, e)$

Primero se ordenan los predicados:

$C_1: \neg p(a, b, c) \vee q(d, c)$

$C_2: \neg p(f, b, e) \vee q(d, e)$

Y ahora se crean los pares: $(a, f), (b, b), (c, e), (d, d), (e, e)$. De acuerdo con el algoritmo anterior, a C_r pasan X, b, Y, d, Y . Así, obtenemos la cláusula más general $\neg p(X, b, Y) \vee q(d, Y)$. C_r es *más general* que C_1 y C_2 si existen las sustituciones θ_1 y θ_2 , tales que $C_r\theta_1 = C_1$ y $C_r\theta_2 = C_2$. Por último, la certidumbre del descendiente puede calcularse mediante el promedio de μ_1 y μ_2 .

Segunda Forma

Si C_1 y C_2 son hechos, y comparten al menos una constante, entonces generar dos nuevas cláusulas: $C_1 \supset C_2$ y $C_2 \supset C_1$. Ambas tendrán certidumbre $(\mu_1 + \mu_2)/4$. Resulta lícito emplear un mayor denominador para obtener una menor certidumbre, considerando que no se dispone de una evidencia sustentable para las relaciones causa-efecto propuestas.

Ahora discutiremos cómo se eligen los pares requeridos. Simplemente se seleccionan al azar dos cláusulas de BC_O y se trata de aplicar alguno de los tipos de emparejamiento descritos. Se acota que si una cláusula descendiente ya se encuentra en la BC, no se agrega, sino que se refuerza la certidumbre de la cláusula existente. Con fines ilustrativos, supóngase que BC_O es:

- (i) *verbo(come)*
- (ii) *predicado(come, arroz)*
- (iii) *verbo(cocina)*
- (iv) *predicado(cocina, arroz)*
- (v) *verbo(estudiar)*
- (vi) *predicado(sirve, pescado)*
- (vii) *verbo(adoba) \supset predicado(adoba, pescado)*
- (viii) *verbo(compra) \supset predicado(compra, pescado)*

El modelo de aprendizaje exige derivar otras BC's a partir de la anterior, mediante los emparejamientos, a fin de que constituyan los individuos de la población inicial. Con una población de tres individuos, puede asumirse que se obtienen:

- Por la segunda forma del emparejamiento inductivo (i y ii), $BC_1 = BC_0 \cup \{ \textit{verbo(come)} \supset \textit{predicado(come, arroz)} [\mu_9 = 0,5] \} \cup \{ \textit{predicado(come, arroz)} \supset \textit{verbo(come)} [\mu_{10} = 0,5] \}$
- Por la primera forma del emparejamiento inductivo (vii y viii), y por deducción (i y xi), $BC_2 = BC_0 \cup \{ \textit{verbo(X)} \supset \textit{predicado(X, pescado)} [\mu_{11} = 0,5], \textit{predicado(come, pescado)} [\mu_{12} = 0,5] \}$
- Por la primera forma del emparejamiento inductivo (vii y viii), y por abducción (xiii y vi), $BC_3 = BC_0 \cup \{ \textit{verbo(X)} \supset \textit{predicado(X, pescado)} [\mu_{13} = 0,5], \textit{verbo(sirve)} [\mu_{14} = 0,333] \}$

En estas condiciones, puede aplicarse el AG. Se requiere la cláusula objetivo, por ejemplo, *verbo(sirve)*. Así, el agente utilizará la evaluación del mejor individuo de la última generación en su proceso de

toma de decisiones. El ejemplo presentado resulta bastante trivial, pero en BC's extensas, las generaciones finales muy probablemente incorporarán la cláusula objetivo, si es posible derivarla.

6. Implementación y Resultados

Resulta claro que la responsabilidad de derivar nuevas cláusulas recae sobre el proceso de emparejamiento. El AG se dedica a retirar de la población aquellas BC's que parecen no conducir al objetivo, usando la heurística representada en la función de evaluación de cláusulas, anteriormente vista. Por tal razón, se requiere un proceso adicional, denominado *promoción del mejor individuo*, en el cual, al término del AG, se extrae de la población final el mejor individuo, y éste se convierte en BC_0 para una nueva aplicación del AG. Al fin, puede presentarse el ciclo del aprendizaje genético:

```
aprendizaje_genetico

// lo que el agente sabe hasta el momento
BC0 <- BC actual del agente

repetir
    1. forma POBLACION a partir de BC0 con emparejamiento
    2. aplica AG sobre POBLACION, hasta alcanzar una determinada
       cantidad de generaciones
    3. BC0 <- mejor individuo de POBLACION final           // promocion

// normalmente, el criterio de terminacion es una cota en las repeticiones
// hasta que se cumpla criterio de terminacion

fin aprendizaje_genetico
```

Al término del ciclo, si el mejor individuo contiene el objetivo (ie., su evaluación excede a 1000.0), el agente puede aceptar que su objetivo es consistente con lo que aprendió previamente. Por ejemplo, el robot concluiría que resulta lícito avanzar hacia el norte, porque percepciones previas incorporadas a BC_0 le han indicado que avance en la dirección despejada, y precisamente hacia el norte no hay obstáculos. Todo lo aprendido puede agregarse a la BC original a fin de usar la nueva información en subsiguientes aplicaciones del AG.

En otro orden de ideas, hay que resaltar que no se encontró software que permitiera trabajar simultáneamente con deducción, abducción

e inducción, en AG, mucho menos usando una representación basada en lógica de predicados, con soporte para incertidumbre. Esto ocasionó que la única opción para verificar el modelo en la práctica fue asumir la compleja labor de la codificación de las ideas presentadas, desde cero. Como lenguaje de implementación se usó C++ porque se requería generalidad para abordar los diversos tipos de inferencia⁹. La implementación posibilitó las siguientes observaciones:

1. La propuesta funciona. Si el objetivo puede obtenerse de la BC original mediante deducción, inducción o abducción, es probable que el AG logre encontrar un individuo suficientemente ajustado, que incluya la cláusula objetivo. También resulta importante apreciar la estrecha relación entre el AG, el proceso de emparejamiento y la heurística codificada en la función de evaluación de cláusulas. En efecto, la función de evaluación guía al AG hacia una BC lo más apta posible para que el emparejamiento obtenga el objetivo. Por su parte, sin el AG, el emparejamiento carece de sentido, pues no sería más que inferencias al azar que no conducirían a ninguna conclusión satisfactoria.
2. No obstante, hay problemas en inferencias extensas y encadenadas, principalmente porque el carácter netamente aleatorio del emparejamiento ocasiona que ciertas derivaciones se pasen por alto. Supóngase que se dispone de la base siguiente¹⁰:

```
not(estadounidense(XA)),not(arma(YA)),not(nacion(ZA)),not(hostil(ZA)),
    ,not(vende(XA,ZA,YA)),delincuente(XA) 1.0
posee(nono,m) 1.0
proyectiles(m) 1.0
not(posee(nono,XC)),not(proyectiles(XC)),vende(west,nono,XC) 1.0
not(proyectiles(XD)),arma(XD) 1.0
not(enemigo(XE,usa)),hostil(XE) 1.0
estadounidense(west) 1.0
nacion(nono) 1.0
enemigo(nono,usa) 1.0
nacion(usa) 1.0
```

Luego, resulta posible inferir el objetivo *delincuente(west)* a partir de esta BC, empleando exactamente 9 pasos deductivos. Sin embargo, al modelo basado en AG le cuesta llegar a dicha conclusión. La Figura 5 muestra las evaluaciones de los mejores individuos en 6 corridas de **aprendizaje_genetico**, con 10 como cota en las repeticiones, y también 10 generaciones en cada ejecución del AG. El tamaño de la población fue 60, y $\alpha_C = 50$. Obsérvese que en sólo 2 se logró alcanzar el objetivo; en general,

⁹ Los lenguajes de programación lógica, como PROLOG, hubiesen resultado convenientes para la inferencia deductiva y la representación de individuos, pero con una dificultad mayúscula para considerar la inducción y la abducción.

¹⁰ Cláusulas extraídas de [35], p. 282.

un α_C grande incrementaría las posibilidades de éxito porque permite trabajar con más cláusulas a la vez.

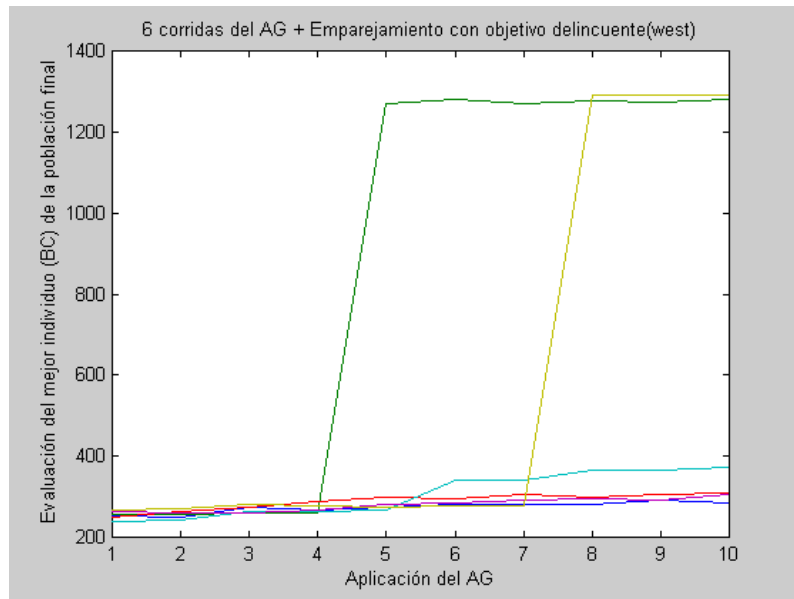


Figura 5. 6 corridas de `aprendizaje_genetico` con objetivo `delincuyente(west)`

3. La representación utilizada, basada en lógica de predicados, consume demasiada memoria, y además el cómputo resulta relativamente lento. Esto se nota claramente con valores altos de α_C y del tamaño de la población.
4. Empíricamente puede comprobarse que, en la mayoría de las ejecuciones de `aprendizaje_genetico`, la cantidad de cláusulas provenientes de la deducción supera a las de los otros mecanismos de inferencia. Tiene sentido si pensamos que los requisitos para aplicar la resolución son menos exigentes que los de la abducción y la inducción, en las formas restringidas que se usan en esta investigación.
5. Por otra parte, el uso simultáneo de deducción y abducción resulta, al menos como se ha planteado, poco útil. Ocurre que la deducción utiliza unas premisas para alcanzar una conclusión, mientras que la abducción parte de la conclusión, e intenta plantear las premisas que la explican. Esto, obviamente, puede ocasionar que el emparejamiento aplique deducción y abducción repetitivamente sobre el mismo conjunto de cláusulas sin

obtener nueva información, desperdiciando el tiempo de cómputo.

7. Conclusiones

El modelo resulta bastante amplio para abordar una gran variedad de problemas, sobre todo aquellos en los que el conocimiento puede representarse fácilmente mediante lógica de primer orden. Luego, empleando el AG se evoluciona la BC, para abarcar más conocimiento. Recuérdese que los agentes raramente accederán a toda la verdad de su ambiente; algunas cláusulas provienen directamente de percepciones, a través de DECIR, mientras que otras se obtienen con INFERIR a partir de las percepciones actuales y previas. En este sentido, el emparejamiento es el encargado de llevar a cabo estas inferencias. El operador de cruce persigue la consecución de una BC consistente con BC_O que permita al emparejamiento derivar el objetivo. La mutación permite explorar otras representaciones sobre el mundo del agente aprendiz. Aquí cabe señalar una posible mejora al modelo: el emparejamiento puede conjuntarse con el cruce genético, a fin de que los dos descendientes incluyan nueva información, no presente en la generación precedente. Esto haría innecesario los procesos de formación de la población inicial y el de promoción, pero con la notoria desventaja de que aplicar emparejamiento en cada cruce enlentecerá demasiado la ejecución del AG, exigirá más memoria, y adicionalmente, se desconoce que tan aptos sean los dos descendientes para merecer que se aplique el emparejamiento en ellos.

La implementación de la propuesta, no detallada aquí, resultó muy difícil y extensa. Básicamente, hay que ocuparse de muchos asuntos a la vez, entre ellos, la representación de cláusulas con un lenguaje no lógico, el algoritmo de unificación, la resolución, la inducción, la abducción, y el manejo de la certidumbre, todo incorporado a los pasos de un AG, susceptibles a la acción de los operadores genéticos, en ocasiones, complementándolo con otros procesos como el emparejamiento y la promoción. Todo debió escribirse desde cero. Se requieren, además, los diccionarios para U_H y U_P . Existen otros detalles críticos para el modelo, a saber:

1. La definición de los emparejamientos (elección de los individuos). Si se definen mal, el emparejamiento prácticamente no tendrá ningún efecto. Adicionalmente, las formas de abducción e inducción utilizadas resultan limitadas, aunque suficientes para esta propuesta. Éstos son tópicos que merecen por sí mismos otra investigación.
2. El cálculo de las certidumbres. Es difícil presentar fórmulas de ajuste de certidumbre lo suficientemente generales para adap-

tarse a todos los dominios. Por ejemplo, en un sistema de diagnósticos médicos, los descendientes del cruce abductivo no deberían penalizarse demasiado a fin de que condujeran a diagnósticos de nivel superior. Por otro lado, el refuerzo que se hace en la certidumbre de cláusulas duplicadas no debe ser muy alto para evitar que por medios extra-lógicos una cláusula obtenga una certidumbre elevada.

También resulta razonable agregar al modelo algún mecanismo de *olvido*, lo cual refleja la realidad de los aprendices naturales, y además permite descartar de la BC aquellas cláusulas que en verdad no aportan nada. Con estos propósitos, cada cierto número de generaciones, se disminuirían las certidumbres de las cláusulas que no intervengan en los emparejamientos (y que tengan una certidumbre menor que 1). Así, por ejemplo, una cláusula resultante de una mutación, que en verdad no aporta nada a la BC, sería eliminada luego de varias generaciones.

Por otra parte, supóngase que a , b y $\neg a \vee \neg b$ pertenecen a la BC, todas con certidumbre menor que 1. Naturalmente, al menos una de las tres cláusulas es falsa. Pero si la falsa es $\neg a \vee \neg b$, no habrá manera de saberlo a priori. Una solución sería incorporar algo así como la *responsabilidad social* de las cláusulas. Se amplía el genotipo a fin de que cada cláusula incluya un enlace a sus padres, y entonces el emparejamiento penalizaría las contradicciones¹¹, siguiendo los enlaces hacia sus padres para penalizar a éstos también. Las penalizaciones continuarán de forma ascendente a los padres de los padres, y así sucesivamente, hasta encontrar cláusulas sin padres, o con certidumbre 1.

Finalmente, el emparejamiento podría ampliarse para que penalizara redundancias como $a \vee \neg a$, y en el tipo inductivo, en su segunda forma, para que aceptara como progenitores una fórmula básica y un hecho, a fin de plantear reglas más complejas.

Debe reconocerse que la representación enlentece el sistema, haciéndolo incapaz, por ahora, de competir con mecanismos específicamente orientados a deducción, abducción o inducción. Presumiblemente, otra forma de representación del conocimiento, de más bajo nivel, podría mejorar el desempeño. Por otro lado, la mutación aporta poco, y en ocasiones se convierte más bien en un factor perturbador. Esto es así porque en cierto sentido, la mutación es *antitológica*, pues altera conocimiento de la BC sin ningún razonamiento como soporte. Debe estudiarse con mucho más detalle el efecto de la mutación sobre esta propuesta de aprendizaje, considerando también la conveniencia de su actuación sobre los predicados en U_P .

¹¹ Cuando el emparejamiento deductivo arroje una cláusula nula.

En definitiva, el aprendizaje empleando las ideas desarrolladas en esta investigación resulta factible, aunque ciertamente se requiere profundizar más en los tópicos presentados.

Referencias

1. Augier, S., Venturini, G., "SIAO1: a first order machine learning system using genetic algorithms", *Proceedings of the ICML96 Workshop on Evolutionary Computing and Machine Learning*, T.C. Fogarty et G. Venturini (Eds), 1996.
2. Augustsson, P., Wolff, K., y Nordin, P., "Creation of a Learning, Flying Robot by means of Evolution", *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1279-1285, 2002.
3. Aurnhammer, M., Tonnies, K., y Mayoral, R., "A Genetic Algorithm for Constrained Seismic Horizon Correlation", *Proceedings of the International Conference on Computer Vision Pattern Recognition and Image Processing (CVPRIP 2002)*, Durham, North Carolina USA, 2002.
4. Bianchi, D. y Delmonte, R., "Tecniche di apprendimento applicate al problema del tagging: una prima valutazione per l'Italiano", *Convegno AI*IA*, 2002.
5. Bruce, J. y Miikkulainen, R., "Evolving Populations of Expert Neural Networks", *Proceedings of the Genetic and Evolutionary Computation Conference - 2001*, pp. 251-257, San Francisco, Morgan Kaufmann, 2002.
6. Cantú-Paz, E., y Chandrika, K., "Evolving Neural Networks for the classification of Galaxies", *Proceedings of the Genetic and Evolutionary Computation Conference - 2002*, pp. 1019-1026, Morgan Kaufmann, 2002.
7. Casillas, J., Cordón, O., del Jesús, M. F., y Herrera, F., "Genetic feature selection in a fuzzy rule-based classification system learning process for high-dimensional problems", *Information Sciences*, Vol. 136, pp. 135-157, 2001.
8. Chakraborti, C., "Human Deductive Reasoning, Genetic Algorithms: A Proposed Model", *Journal of Indian Council of Philosophical Research*, Vol. 18, pp. 61-90, 2001.
9. Cordón, O., Herrera, F., del Jesus, M. F., Magdalena, L., Sánchez, A., y Villar, P., "A Multiobjective Genetic Algorithm for Feature Selection and Data Base Learning in Fuzzy-Rule Based Classification Systems", *Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems*, 2002.
10. Dagupta, D. y González, F., "Evolving complex fuzzy classifier rules using a linear genetic representation", *International Conference Genetic and Evolutionary Computation - 2001*, 2001.
11. Dass, M., Cannady, J., y Don Potter, W., "A GA based Intrusion Sub-classifier Filter", *ACM Southeast Regional Conference*, Georgia, 2003.

12. DiPietro, A., y While, L., y Barone, L., "Learning In RoboCup Keep-away Using Evolutionary Algorithms", *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1065-1072, 2002.
13. Echizenya, H. et al, "Application of Genetic Algorithms for Example-Based Machine Translation Method Using Inductive Learning and Its Effectiveness", *IPSJ JOURNAL Vol. 37*, 2001.
14. Foro electrónico. "How might you use genetic algorithms to produce a form of intelligent program?" [URL: <http://www.math.grin.edu/rebel-sky/Courses/CS105/2000S/Questions/question.38.html>]
15. Galán, S., Velasco, J., y Magdalena, L., "Evolutionary Learning Through Growth of the Domain Coverage Using a Classifier System", *Proceedings of the 5th WSES Multiconference on Circuits, Systems, Communications y Computers*, 2001.
16. Gomez et al, "Complete Expression Trees for Evolving Fuzzy Classifier Systems with Genetic Algorithms And Application to Network intrusion Detection", *Proceedings of NAFIPS-FLINT joint conference*, 2002.
17. Hamel, L., "Breeding Algebraic Structures: An Evolutionary Approach To Inductive Equational Logic Programming", *International Conference Genetic and Evolutionary Computation - 2002*, pp. 748-755, 2002.
18. Hancock, P., y Frowd, C., "Evolutionary generation of faces", *Creative Evolutionary Systems*, Morgan Kaufmann, 2002.
19. Joglekar, A., y Tungare, M., "Genetic Algorithms and their use in the design of Evolvable Hardware", [URL: <http://www.manastungare.com/articles/default.asp>], 2002.
20. Kim, S., y Zhang, B., "Web document retrieval by genetic learning of importance factors for html tags", *Proc. Int. Workshop Text Web Mining*, Australia, pp. 13-23, 2000.
21. Kosorukoff, A., y Goldberg, D., "Genetic Algorithms for Social Innovation and Creativity", *IlIiGAL Report No. 2001005*, Illinois, 2001.
22. Kosorukoff, A., y Goldberg, D., "Evolutionary computation as a form of organization", *Proceedings of the Genetic and Evolutionary Computation Conference - 2002*, 2002.
23. Levitsky, V., y Katokhin, A., "Recognition of eukaryotic promoters using a genetic algorithm based on iterative discriminant analysis", *Silico Biology*, Vol. 3, 2003.
24. Manzolli, J., Moroni, A., Von Zuben, F., Gudwin, R., "An Evolutionary Approach Applied to Algorithmic Composition", *Proceedings of SBC'99 - XIX National Congress of the Computation Brazilian Society*, Vol. 3, 1999.
25. McCormack, J., "Evolving Sonic Ecosystems", *Kybernetes Vol. 32 (1/2)*, 2002.
26. McQuesten, P., "Cultural Enhancement of Neuroevolution", *Department of Computer Science, University of Texas*, Ph.D. Thesis, 2002.
27. Mitchell, T., *Machine Learning*, pp. 1-3, McGraw-Hill, 1997.
28. Moffat, K., y Fels, S., "Interactive Genetically Generated Music: Composing with a Genetic Algorithm", [URL: <http://taz.cs.ubc.ca/egems/karyn/projects/2001/iggm.html>], 2001.

29. Moroni, A., Manzolli, J., Von Zuben, F., "Composing with Interactive Genetic Algorithms", [URL: http://www.nics.unicamp.br/papers_nics/composing_interactive.pdf], 2001.
30. Muggleton, S. y De Raedt, L., "Inductive Logic Programming: Theory and Methods", *Journal of Logic Programming*, 1994.
31. Neri, F., y Giordana, A., "A Distributed Genetic Algorithm for Concept Learning", *Proc. Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, pp. 436-443, 1995.
32. Neri, F., y Saitta, L., "An Analysis of the Universal Suffrage Selection Operator", *Evolutionary Computation Journal*, MIT Press, vol. 4, n. 1, pp. 89-109, 1997.
33. Prerau, M., "On the Possibilities of an Analytic Synthesis System Using Construct Based Genetic Algorithms", *Proceedings of the European Conference on Artificial Life 2000 workshop: Artificial Life Models for Musical Applications*, Praga, 2001.
34. Rich, E., y Knight, K., "Inteligencia Artificial", McGraw-Hill, 1996.
35. Russell, S., y Norvig, P., "Inteligencia Artificial", Prentice Hall Hispanoamericana, 1996.
36. Sasaki, Y., "Can Evolvable AI Interpret Satellite Imagery?", [URL: <http://cc.usu.edu/slk1r/technology/genetclassif/>], 2002.
37. Stanley, K. O. y Miikkulainen, R., "Evolving Neural Networks through Augmenting Topologies", *Evolutionary Computation*, 10(2):99-127, 2002.
38. Stanley, K. O. y Miikkulainen, R., "Efficient Reinforcement Learning through Evolving Neural Network Topologies", *Proceedings of the Genetic and Evolutionary Computation Conference - 2002*, San Francisco, CA: Morgan Kaufmann, 2002.
39. Stanley, K. O. y Miikkulainen, R., "Achieving High-Level Functionality through Complexification", *Proceedings of the AAAI-2003 Spring Symposium on Computational Synthesis*, Stanford, CA: AAAI Press, 2003.
40. Stanley, K. O. y Miikkulainen, R., "Competitive Coevolution through Evolutionary Complexification", *Tech Report AI-02-298, Department of Computer Sciences, University of Texas*, 2002.
41. Stanley, K. O. y Miikkulainen, R., "Continual Coevolution through Complexification", *Proceedings of the Genetic and Evolutionary Computation Conference - 2002*, San Francisco, CA: Morgan Kaufmann, 2002.
42. Stanley, K. O. y Miikkulainen, R., "Efficient Evolution of Neural Network Topologies", *Proceedings of the 2002 Congress on Evolutionary Computation (CEC '02)*, Piscataway, NJ: IEEE, 2002.
43. Wirth, U., "What is Abductive Inference", *Encyclopedia of Semiotics*, Oxford University Press, 1998.
44. Yabuki, T., y Hitoshi, I., "Genetic Algorithms for Quantum Circuit Design: Evolving a Simpler Teleportation Circuit", *Genetic and Evolutionary Computation Conference-2000*, pp. 425-430, 2000.
45. Zhou, Z.-H., y Chen, S.-F., "Evolving Fault-Tolerant Neural Networks", *Neural Computing and Applications*, 2003.